REMARKS/ARGUMENTS

Reconsideration of this application is respectfully requested.

The rejection of claims 17-23, 25-29, 31-33, 35-37, 39, 40, and 42-46 under 35 U.S.C. § 102 as allegedly anticipated by Filo '498 is respectfully traversed.

Filo teaches a virtual command post so that the commanding officer (the executive level user) and soldiers in the field (the operations level users) are displayed in a virtual environment (that also depicts lower level staff level users who are not outfitted with position sensors). The two higher level users (the commanding officer and soldiers) wear body-part position sensors (e.g. a head tracker 38 and a hand/finger activated button 46) so as to control the displayed virtual environment and/or communicate with one another via such potential signaling mechanisms.

However, Filo does not teach any means for detecting environmental or physical attributes that impair the ability of a user to continue communication in a currently occurring mode—coupled with means for automatically changing the user's mode of communication so as to accommodate the detected impairment.

For example, in applicants' claimed invention, a mobile user device may detect increased ambient noise which then automatically changes the user's mode of communication (e.g. by increasing the audio output volume to the mobile user) and also causes the representation of that user to the at least one other party to change in a manner that reflects this changed mode of communication.

Another example of use for the applicants' invention may be detection of the user in running motion whereupon a communication mode changes from visual to audio. Similarly, the applicants' invention could be used for detecting user motion in a vehicle (e.g. via velocity of movement as reflected by a GPS receiver or the like) and automatically switch from a visual to an audio mode of communication (and, of course, concurrently causing the at least one other party to the communication to be presented with a changed representation of that user representing such changed mode of communication).

All independent claims have been amended above so as to require detection of an environmental or physical attribute impairing the ability of the user to continue communication in a currently occurring mode and then to automatically change the user's mode of communication so as to accommodate the detected impairment while also simultaneously modifying the user's representation in the virtual environment so as to reflect this detected environmental or physical attribute. Such amendments therefore moot this outstanding ground of rejection – even if it is assumed *arguendo* that the body-part tracking means 38, 46 in Filo detect something that affects communication of the user (which, if it does, is not mentioned to do so by Filo).

The rejection of claims 24, 30 and 34 under 35 U.S.C. § 103 as allegedly being made "obvious" based on Filo in combination with Nitta '306 is also respectfully traversed.

The fundamental deficiencies of Filo have already been noted above with respect to all independent claims. Nitta does not supply those deficiencies.

Further, Nitta appears to be directed towards improving a virtual environment depiction by causing stereophonic sound generation to accompany the virtual representation environment – thus making it seem as if the accompanying sound emanates from the animation representation of the person supposedly currently speaking. Accordingly, this teaching appears to be essentially irrelevant to even the incremental aspects of the applicants' invention added by these dependent claims.

Indeed, the Examiner's comments with respect to claim 34 indicates a possible lack of understanding with respect to what constitutes a "Hidden Markov Model". It is noted by the applicants in the paragraph bridging pages 8 and 9 of the specification, that some embodiments of the applicants' invention may find it useful to employ a Hidden Markov Model approach to detect the particular environment or physical attribute that one is trying to detect. This is because, at least for some physical attributes or environmental attributes, the available raw physical data representing such may be quite complex involving multiple variables which, individually, may provide ambiguous information. Hidden Markov Models are, for example, often used in speech recognition algorithms and the like for similar reasons.

Attached hereto for the Examiner's convenience, is a printout obtained via Google from Wikipedia, the free encyclopedia. This four-page description of a Hidden Markov Model may assist the Examiner in understanding that such is not at all suggested by anything in Filo or any allegedly "hidden state system" in Nitta.

The passages cited by the Examiner in Nitta at columns 3 and 6 are alleged to be related to estimating the position of a user that has an unknown exact position (and perhaps to that extent a "hidden" position). Actually, the portions of text cited by the Examiner do not appear to support the Examiner's allegations. Nevertheless, even if such is assumed *arguendo* to be present in the Nitta teachings, this is not at all any teaching or suggestion of Hidden Markov Model processing of data indicative of environmental or physical attributes, etc. as specified in applicants' claim 34.

The rejection of claims 38 and 41 under 35 U.S.C. § 103 as allegedly being made "obvious" based on Filo in view of Sun '740 is also respectfully traversed.

Once again, fundamental deficiencies of Filo have already been noted above with respect to parent claims. Sun does not supply those deficiencies.

Furthermore, even though the Sun system apparently automatically recognizes the type of connection available for each conferee and then effects that type of available connection, it does not appear to anywhere suggest detecting a quality of service level of the connection. As is well known in the industry, there is a difference between a "type of connection" (e.g. full duplex, half duplex, etc. as depicted in the spreadsheet graphical user interface at figure 6 of Sun) and the "quality of service level" of a given connection (e.g., bandwidth, regardless of its "type").

In any event, claims 38 and 41 require configuring or modifying a representation of the user in dependence upon the detected quality of service of the connection. The brief mention at column 5, line 61-63 to the effect that "conferees may be connected using the present invention in other modes, including virtual reality animation, graphics and text" does not teach or suggest that a representation of the user being provided to the other conference parties would be modified in dependence upon the detected "quality of service" of the connection.

Accordingly, this entire application is now believed to be in condition for allowance and a formal notice to that effect is respectfully solicited.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: Larry S. Nixon
Larry S. Nixon
Reg. No. 25,640

LSN:dm
901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100

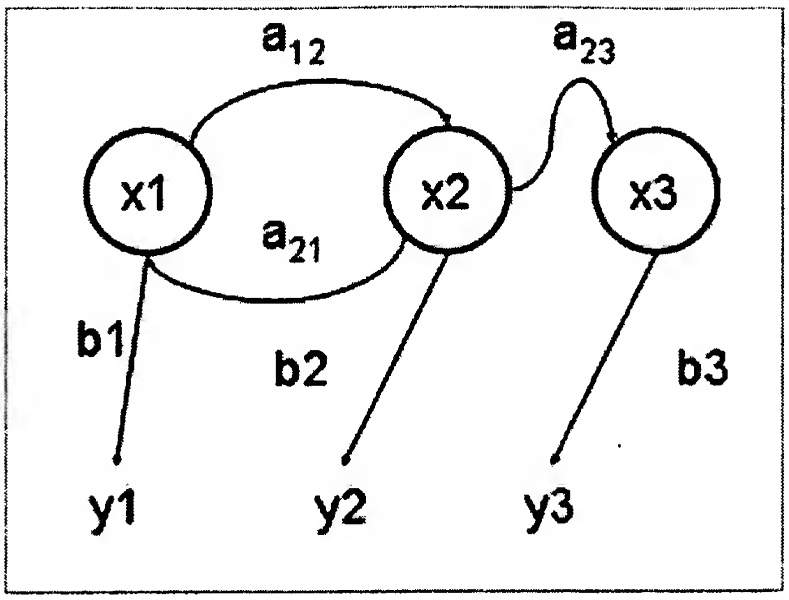
Hidden Markov model

From Wikipedia, the free encyclopedia

A **hidden Markov model** (HMM) is a statistical model where the system being modeled is assumed to be a Markov process with unknown parameters, and the challenge is to determine the hidden parameters from the observable parameters. The extracted model parameters can then be used to perform further analysis, for example for pattern recognition applications.

In a regular Markov model, the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters. In a *hidden* Markov model, the state is not directly visible, but variables influenced by the state are visible. Each state has a probability distribution over the possible output tokens. Therefore the sequence of tokens generated by an HMM gives some information about the sequence of states.

Hidden Markov models are especially known for their application in speech recognition and biological sequence analysis (bioinformatics).

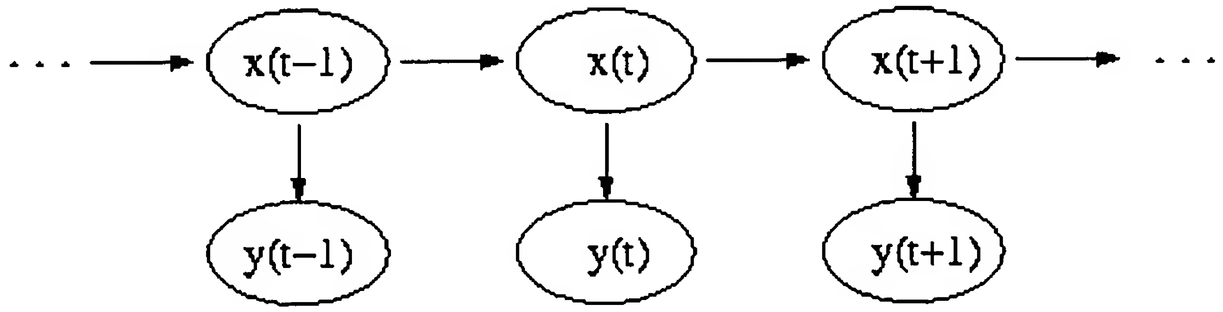


State transitions in a hidden Markov model (example)
x — hidden states
y — observable outputs
a — transition probabilities
b — output probabilities

Contents

Evolution of a Markov model

The preceding diagram emphasizes the state transitions of a HMM. It is also useful to explicitly represent the evolution of the model over time, with the states at different times t_1 and t_2 represented by different variables, $x(t_1)$ and $x(t_2)$.



In this diagram, it is understood that the time slices ($x(t)$, $y(t)$) extend to previous and following times as needed. Typically the earliest slice is at time t=0 or time t=1.

Using Markov models

There are 3 canonical problems to solve with HMMs:

- Given the model parameters, compute the probability of a particular output sequence. Solved by the forward algorithm.
- Given the model parameters, find the most likely sequence of (hidden) states which could have generated a given output sequence. Solved by the Viterbi algorithm.
- Given an output sequence, find the most likely set of state transition and output probabilities. Solved by the Baum-Welch algorithm or the Reversed Viterbi algorithm.

Another, more recent approach is to solve these problems by using the Junction tree algorithm.

A concrete example

Assume you have a friend who lives far away and to whom you talk daily over the telephone about what he did that day. Your friend is only interested in three activities: walking in the park, shopping, and cleaning his apartment. The choice of what to do is determined exclusively by the weather on a given day. You have no definite information about the weather where your friend lives, but you know general trends. Based on what he tells you he did each day, you try to guess what the weather must have been like.

You believe that the weather operates as a discrete Markov chain. There are two states, "Rainy" and "Sunny", but you cannot observe them directly, that is, they are *hidden* from you. On each day, there is a certain chance that your friend will perform one of the following activities, depending on the weather: "walk", "shop", or "clean". Since your friend tells you about his activities, those are the *observations*. The entire system is that of a hidden Markov model (HMM).

You know the general weather trends in the area, and what your friend likes to do on average. In other words, the parameters of the HMM are known. You can write them down in the Python programming language:

```
states = ('Rainy', 'Sunny')

observations = ('walk', 'shop', 'clean')

start_probability = {'Rainy': 0.6, 'Sunny': 0.4}

transition_probability = {
   'Rainy' : {'Rainy': 0.7, 'Sunny': 0.3},
   'Sunny' : {'Rainy': 0.4, 'Sunny': 0.6},
   }

emission_probability = {
   'Rainy' : {'walk': 0.1, 'shop': 0.4, 'clean': 0.5},
   'Sunny' : {'walk': 0.6, 'shop': 0.3, 'clean': 0.1},
   }
```

In this piece of code, `start_probability` represents your uncertainty about which state the HMM is in when your friend first calls you (all you know is that it tends to be rainy on average). The particular probability distribution used here is not the equilibrium one, which is (given the transition probabilities) actually approximately `{'Rainy': 0.571, 'Sunny': 0.429}`. The `transition_probability` represents the change of the weather in the underlying Markov chain. In this example, there is only a 30% chance that tomorrow will be sunny if today is rainy. The `emission_probability` represents how likely your friend is to perform a certain activity on each day. If it is rainy, there is a 50% chance that he is cleaning his apartment; if it is sunny, there is a 60% chance that he is outside for a walk.

This example is further elaborated in Viterbi algorithm page. g

Applications of hidden Markov models

- speech recognition or optical character recognition
- machine translation
- bioinformatics and genomics
    - prediction of protein-coding regions in genome sequences
    - modelling families of related DNA or protein sequences
    - prediction of secondary structure elements from protein primary sequences
- *and many more...*

History

Hidden Markov Models were first described in a series of statistical papers by Leonard E. Baum and other authors in the second half of the 1960s. One of the first applications of HMMs was speech recognition, starting in the mid-1970s.[1]

In the second half of the 1980s, HMMs began to be applied to the analysis of biological sequences, in particular DNA. Since then, they have become ubiquitous in the field of bioinformatics.[2]

See also

- Andrey Markov
- Baum Welch algorithm
- Bayesian inference
- estimation theory
- Viterbi algorithm
- Hierarchical hidden Markov model
- Hidden semi-Markov model

Notes

1. ^ Rabiner, p. 258
2. ^ Durbin et al.

References

- Lawrence R. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition (http://www.caip.rutgers.edu/~lrr/Reprints/tutorial%20on%20hmm%20and%20applications.pdf). *Proceedings of the IEEE*, 77 (2), p. 257–286, February 1989.
- Richard Durbin, Sean R. Eddy, Anders Krogh, Graeme Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1999. ISBN 0521629713.
- Lior Pachter and Bernd Sturmfels. "Algebraic Statistics for Computational Biology" Cambridge University

Press, 2005. ISBN 0521857007

- Kristie Seymore, Andrew McCallum, and Roni Rosenfeld. *Learning Hidden Markov Model Structure for Information Extraction.* AAAI 99 Workshop on Machine Learning for Information Extraction, 1999. *(also at CiteSeer: [1] (http://citeseer.ist.psu.edu/seymore99learning.html))*
- http://www.comp.leeds.ac.uk/roger/HiddenMarkovModels/html_dev/main.html
- J. Li (http://www.stat.psu.edu/~jiali), A. Najmi, R. M. Gray, Image classification by a two dimensional hidden Markov model, *IEEE Transactions on Signal Processing*, 48(2):517-33, February 2000.

External links

- Hidden Markov Model (HMM) Toolbox for Matlab (http://www.ai.mit.edu/~murphyk/Software/HMM/hmm.html) *(by Kevin Murphy)*
- Hidden Markov Model Toolkit (HTK) (http://htk.eng.cam.ac.uk/) *(a portable toolkit for building and manipulating hidden Markov models)*
- Hidden Markov Models (http://www.cs.brown.edu/research/ai/dynamics/tutorial/Documents/HiddenMarkovModels.html) *(an exposition using basic mathematics)*
- GHMM Library (http://www.ghmm.org/) *(home page of the GHMM Library project)*
- Jahmm Java Library (http://www.run.montefiore.ulg.ac.be/~francois/software/jahmm/) *(Java library and associated graphical application)*
- A step-by-step tutorial on HMMs (http://www.comp.leeds.ac.uk/roger/HiddenMarkovModels/html_dev/main.html) *(University of Leeds)*
- Software for Markov Models and Processes (http://www.treeage.com/products/overviewHealth.html) *(TreeAge Software)*

Retrieved from "http://en.wikipedia.org/wiki/Hidden_Markov_model"

Categories: Statistics | Machine learning | Computer vision

- This page was last modified 10:58, 24 July 2006.
- All text is available under the terms of the GNU Free Documentation License. (See **Copyrights** for details.) Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc.